04-18-05

AF/ IFW 2155

ATTORNEY DOCKET NO.
073388.0122

PATENT APPLICATION
09/451,699

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

| | |
|---|---|
| In re Application of: | Chris K. Wensel |
| Serial No.: | 09/451,699 |
| Filing Date: | November 30, 1999 |
| Group Art Unit: | 2155 |
| Examiner: | Michael Young Won |
| Title: | TRANSPARENT CONNECTION TYPE BINDING BY ADDRESS RANGE |

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

## CERTIFICATE OF MAILING BY EXPRESS MAIL

I hereby certify that the attached Appeal Brief with check is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 C.F.R. §1.10 on this 14th day of April 2005, addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

Willie Jiles
Willie Jiles

Express Mail Receipt
No. EV 322269061 US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of: Chris K. Wensel
Serial No.: 09/451,699
Filing Date: November 30, 1999
Group Art Unit: 2155
Examiner: Michael Young Won
Title: TRANSPARENT CONNECTION TYPE BINDING BY ADDRESS RANGE

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

## APPEAL BRIEF

Applicant has appealed to the Board of Patent Appeals and Interferences from the decision of the Examiner mailed October 13, 2004, finally rejecting Claims 1-27. Applicant filed a Notice of Appeal on February 14, 2005. Applicant respectfully submits herewith their brief on appeal with a statutory fee of $250.00.

## REAL PARTY IN INTEREST

The present Application was assigned to Objectspace, Inc., a Delaware corporation, as indicated by an assignment from the inventor recorded on November 30, 1999 in the Assignment Records of the United States Patent and Trademark Office at Reel 010423, Frames 0632-0634. The present Application was subsequently assigned to Recursion Software, Inc., a Texas corporation, as indicated by an assignment from Objectspace, Inc. recorded on February 22, 2002 in the Assignment Records of the United States Patent and Trademark Office at Reel 012588, Frames 0215-0218.

## RELATED APPEALS AND INTERFERENCES

There are no known appeals or interferences that will directly affect or be directly affected by or have a bearing on the Board's decision in this pending appeal.

## STATUS OF CLAIMS

Claims 1-27 stand rejected pursuant to a Final Action mailed October 13, 2004. Claims 1-27 are all presented for appeal.

## STATUS OF AMENDMENTS

A Response to Examiner's Final Action was filed on December 13, 2004 in response to the Final Action mailed October 13, 2004. No amendments were made to the claims. The Examiner issued an Advisory Action dated January 26, 2005 which stated that the Response to Examiner’s Final Action was considered but that it did not place the application in condition for allowance. A Notice of Appeal was filed on February 14, 2005.

## SUMMARY OF CLAIMED SUBJECT MATTER

With reference to Figures 1-3 and Applicant's specification, the present invention provided in Claims 1, 10, 12, and 23 involves a system and method for traversing a boundary in a distributed processing environment that includes a connection properties table (80) in a client network (18) that stores connection protocol information (82, 84, 86, 88) for each boundary (28) of a server network (26) that the client network (18) may traverse. A client object on a client machine (16) requests access (110) to a server object on a server machine (24) in the server network (26). An entry is located (130) in the connections property table (80) of the client network (18) that corresponds to the requested server object. A boundary traversal key is formed (160) from the connection protocol information associated with the entry in the connection properties table (80). The boundary traversal key includes information to traverse the boundary (28) controlling access to the server network (26). The request for access and the boundary traversal key are forwarded (170) to the boundary (28) controlling access to the server network (26) for processing to allow communications (190) between the client network (18) and the server network (26). As provided in Claims 14, 24, and 26, the client network (18) may communicate with the server network (26) through an intermediate network (22) (Page 9, line 32, to Page 10, line 1). As provided in Claim 18, the connection properties table (80) may be stored in a private directory of the client network (18) (Page 14, lines 10-12). As provided in Claims 3, 4, 15, 16, 25, and 27, an object request broker (30) may process the request for access from the client network (18), form the boundary traversal key, and forward the request and boundary traversal key towards the server network (26) (Page

12, line 11, to page 13, line 3). As provided in Claims 2, 5, 17, and 23, the connection protocol information may include a boundary identifier (82), a connection type (84), authentication information (86), and connection attributes (88) (Page 13, lines 9-15). As provided in Claim 22, the authentication information (86) may be a user identification and a password (Page 13, lines 30-32). As provided in Claim 21, connections may be TCP/IP, SSL, HTTP Tunneling, or UDP/IP (Page 13, lines 16-24). As provided in Claims 6-8 and 20, an entry in the connections property table may be identified by matching of an internet protocol address, an internet protocol address range, a partial internet protocol address, a domain name, a partial domain name, a port address, or a port address range for the requested server object (Page 15, lines 26-30). As provided in Claim 9 and 19, the boundary traversal key is formatted with authentication information (86) and attributes (88) as defined by the connection type (84) (Page 16, line 30, to Page 17, line 3). As provided in Claim 11, the server network boundary (28) receives the access request and boundary traversal key and either allows or denies access to the server object (Page 11, lines 14-18, and Page 12, lines 24-27). As provided in Claim 13, if no entry is found (150) in the connection properties table (80), a default connection type is used (Page 16, lines 17-23).

## GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

1. Did the Examiner err in concluding that Claims 1-27 were anticipated under 35 U.S.C. §102(e) by U. S. Patent No. 6,5,876,665 issued to Butman, et al.?

## ARGUMENT

1. Claims 1-27 stand rejected under 35 U.S.C. §102(e) as being anticipated by Butman, et al. To anticipate a claim under 35 U.S.C. §102(b), a single prior art reference must teach each and every limitation as set forth in the claim. Since the cited prior art reference does not teach each and every element set forth in the claims, Applicant respectfully traverses this rejection.

Claims 1-27 stand rejected under 35 U.S.C. §102(e) as being anticipated by Butman, et al. Independent Claims 1, 12, and 23 recite in general the formation of a boundary traversal key from a connections property table at a client network where the connections property table includes information to traverse a boundary device controlling access to a server network in response to a desire by a client object in the client network to access a server object in the server network. By contrast, the Butman, et al. patent performs its authentication process for the client side communications servers at a domain communications server through use of a dynamic client registry maintained at the domain communications server. Thus, there is no connection property table at the client side communications server of the Butman, et al. patent as required by the claimed invention.

Moreover, the domain communications server of the Butman, et al. patent collects information for a particular client side communication server from other client side communications server and controls communications between client side communications servers. See col. 27, lines 26-32, of the Butman, et al. patent. Thus, the particular client side communications server of the Butman, et al. patent is never required to generate a boundary traversal key in order to traverse a boundary device controlling access to a domain communications server or another client side communications server and thus does not create a boundary traversal key as required by the claimed invention.

The portions of the Butman, et al. patent cited by the Examiner and shown in Figure 3 merely illustrates an extension of Figure 1 where domain communications servers are capable of being connected to each other. The connection of domain communications servers to one another merely provides redundant operation in the event of a failure of a particular domain communications server. See col. 15, lines 6-18, of the Butman, et al. patent. There is no disclosure in the Butman, et al. patent with respect to any boundary between two domain communications servers let alone for one domain communications server to generate a boundary traversal key in order to gain access to another domain communications server.

The Examiner has yet to show any disclosure in the Butman, et al. patent that the client side communication servers connected to these respective domain communications servers include a connection property table as required in the claimed invention or that any boundary traversal key is generated. The only disclosure provided by the Butman, et al. patent and cited by the Examiner is of a dynamic client registry maintained at the domain communications servers to authenticate the client side communications servers. The domain communications server does not even generate anything resembling the claimed boundary traversal key from its dynamic client registry. The domain communications servers clearly cannot be considered "client side" as they are shown to be separated from the client side communications servers by individual firewalls. Thus, not only do the client side communications servers of the Butman, et al. patent fail to have the claimed connection properties table, there is also no capability for the client side communications servers to generate a boundary traversal key. In simple terms, the claimed invention provides for a client desiring access to a server to generate a boundary traversal key from information at the client in order to unlock the appropriate door at the server and satisfy its desire for access. There is nothing in the Butman, et al. patent that has any server make its own key to gain entry through another server's door.

According to the above discussion, Independent Claims 1, 12, and 23, as well as their dependent claims, are not anticipated by the Butman, et al. patent.

## CONCLUSION

Applicant has clearly demonstrated that the present invention as claimed is clearly distinguishable over all the art cited of record, either alone or in combination, and satisfies all requirements under 35 U.S.C. §§101, 102, and 103, and 112. Therefore, Applicant respectfully requests the Board of Patent Appeals and Interferences to reverse the final rejection of the Examiner and instruct the Examiner to issue a notice of allowance of all claims.

Attached herewith is a check made payable to the "Commissioner of Patents and Trademarks" in an amount of $250.00 in order to satisfy the appeal brief filing fee of 37 C.F.R. §41.20(b)(2).

The Commissioner is hereby authorized to charge any fees or credit any overpayments to Deposit Account No. 02-0384 of BAKER BOTTS L.L.P.

Respectfully submitted,
BAKER BOTTS L.L.P.
Attorneys for Applicant

Charles S. Fish
Reg. No. 35,870

April 14, 2004

Correspondence Address:
2001 Ross Avenue, Suite 600
Dallas, TX 75201-2980
(214) 953-6507
Customer Number: 05073

## APPENDIX A

1. (Previously Presented) A method for traversing a boundary in a distributed processing environment, comprising:

storing connection protocol information in a connection properties table at a client network for each boundary which may be traversed by the client network;

receiving a request from a client object on the client network for access to a server object on a server network, the server network having a server network boundary;

locating an entry in the connections property table corresponding to the requested server object;

formatting a boundary traversal key from the connection protocol information associated with the located entry in the connection properties table, the boundary traversal key including information to traverse a boundary controlling access to the server network; and

forwarding the request for access and the boundary traversal key to the boundary controlling access to the server network.

2. (Original) The method of Claim 1, further comprising determining a connection type from the located entry in the connections property table.

3. (Original) The method of Claim 1, further comprising:

passing the request for access to an object request broker after the client network determines that the request for access is to an object residing outside the client network.

4. (Original) The method of Claim 3, wherein the object request broker locates the entry, formats the boundary traversal key, and forwards the request for access and the boundary traversal key to the server network.

5. (Original) The method of Claim 1, wherein storing connection protocol information includes storing a boundary identifier, a connection type, authentication information, and connection attributes in the connection properties table.

6. (Original) The method of Claim 5, wherein locating an entry includes matching an internet protocol address for the server object to the boundary identifiers stored in the connection properties table.

7. (Original) The method of Claim 5, wherein locating an entry includes matching a domain name for the server object to the boundary identifiers stored in the connection properties table.

8. (Original) The method of Claim 5, wherein locating an entry includes matching a port address for the server object to the boundary identifiers stored in the connection properties table.

9. (Original) The method of Claim 5, wherein formatting the boundary traversal key includes building the boundary traversal key from the authentication information and the connection attributes in a format defined by the connection type.

10. (Original) The method of Claim 1, wherein forwarding the request includes forwarding the request for access and the boundary traversal key to the server network boundary.

11. (Original) The method of Claim 1, further comprising:

receiving the request for access and the boundary traversal key at the server network boundary;

allowing access to the server object if the server network boundary accepts the boundary traversal key; and

denying access to the server object if the server network boundary rejects the boundary traversal key.

12. (Previously Presented) A distributed computing system, comprising:

a client object on a first network operable to request access to a server object on a second network;

a third network connecting the first network to the second network;

a boundary device controlling access to the second network;

a connections properties table in the first network and including an entry for each of one or more second networks accessible by the first network, the connections properties table including connection protocol information for accessing the one or more second networks;

a connection manager operable to generate a boundary traversal key for requests for access to server objects that have a corresponding entry in the connections properties table, the boundary traversal key generated from the corresponding connection protocol information, the boundary traversal key including information to traverse the boundary device controlling access to the second network.

13. (Original) The system of Claim 12, further comprising a default connection manager operable to establish a connection between the client object and the server object using a default protocol for requests for access to server objects that do not have a corresponding entry in the connection properties table.

14. (Original) The system of Claim 12, wherein the third network is an Internet.

15. (Original) The system of Claim 12, further comprising an object request broker operable to facilitate communications between the client object and the server object across the third network.

16. (Original) The system of Claim 15, wherein the connection manager is part of the object request broker.

17. (Original) The system of Claim 12, wherein the connection properties table includes:

a boundary identifier for identifying the server object on the second network;

a connection type for identifying the type of connection protocol used by the second network;

authentication information for providing identity and credential information to the second network; and

attributes for providing boundary traversal key information to the second network.

18. (Original) The system of Claim 12, wherein the connection properties table is stored in a private directory on the first network.

19. (Original) The system of Claim 17, wherein the boundary traversal key is generated from the authentication information and the attributes from an entry in the connection properties table corresponding to the server object on the second network.

20. (Original) The system of Claim 17, wherein the boundary identifier is an identifier selected from the group consisting of an internet protocol address, an internet protocol address range, a partial internet protocol address, a domain name, a partial domain name, a port address and a port address range.

21. (Original) The system of Claim 17, wherein the connection type indicates a TCP/IP connection, an SSL connection, an HTTP Tunneling connection, or a UDP/IP connection.

22. (Original) The system of Claim 17, wherein the authentication information includes a user identification and a password.

23. (Previously Presented) A distributed processing system with transparent boundary traversal, comprising:

a client system operable to request access to a plurality of server systems, at least one of the server systems having a boundary device for controlling access to the server system by the client system;

a connection properties table stored in a private directory on the client system, the connection properties table including:

an identification range for identifying the at least one server system having the boundary device;

a boundary type for identifying a type of the boundary device;

authentication information for uniquely identifying the client system to the boundary device and a requested server system; and

attributes for providing traversal information required by the boundary device;

a boundary traversal key generator operable to generate a boundary traversal key for gaining access to the requested server system through the boundary device, the boundary traversal key generated from the connection properties table in response to the boundary traversal key generator locating an entry matching the requested server system.

24. (Original) The system of Claim 23, further comprising a network for connecting the client system to the server system.

25. (Original) The system of Claim 24, further comprising an object request broker operable to facilitate communications between the client object and the server object across the network.

26. (Original) The system of Claim 24, wherein the network is an internet.

27. (Original) The system of Claim 25, wherein the boundary traversal key generator is part of the object request broker.